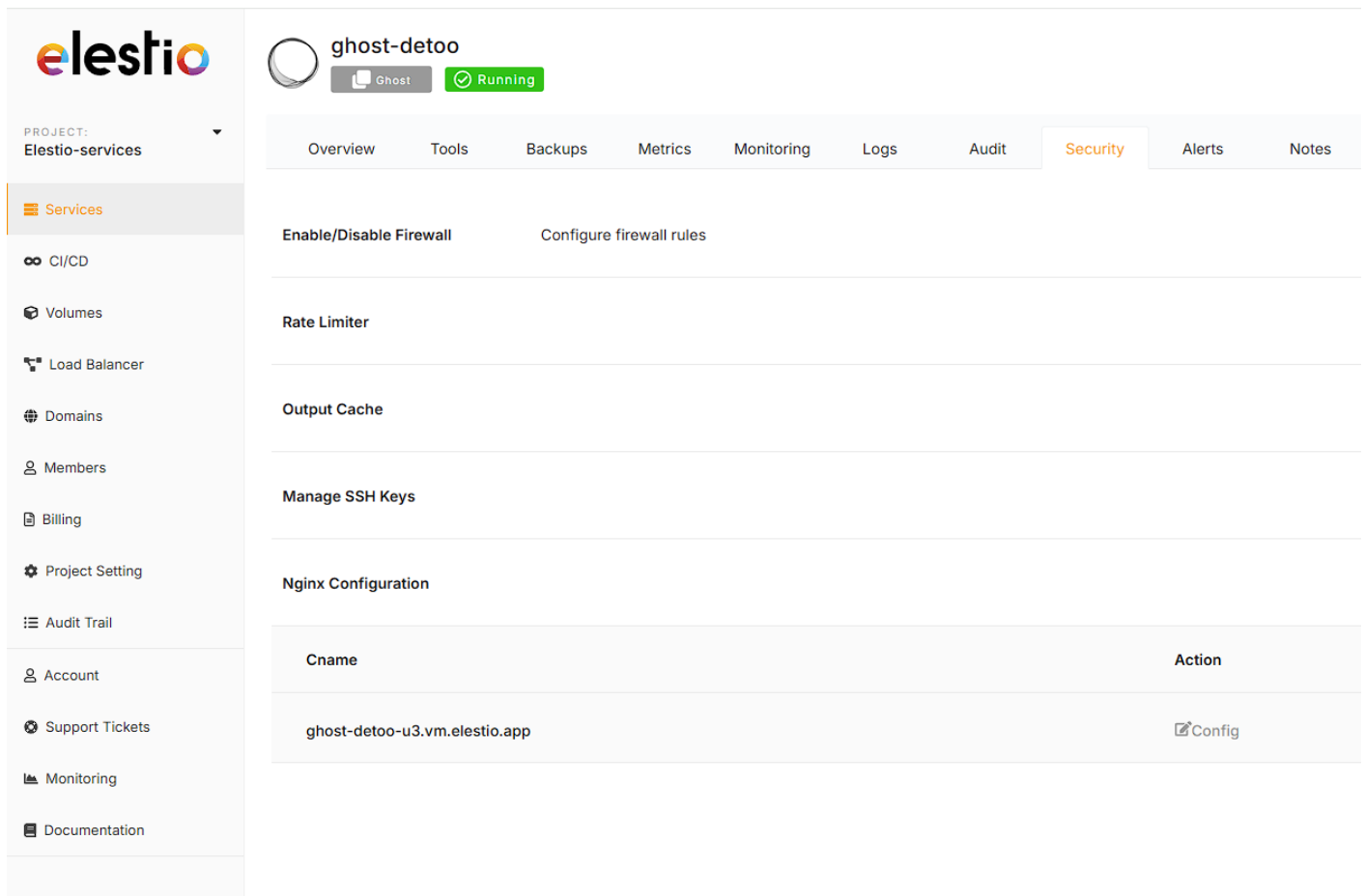


# Nginx advanced configuration

## Edit the website nginx configuration

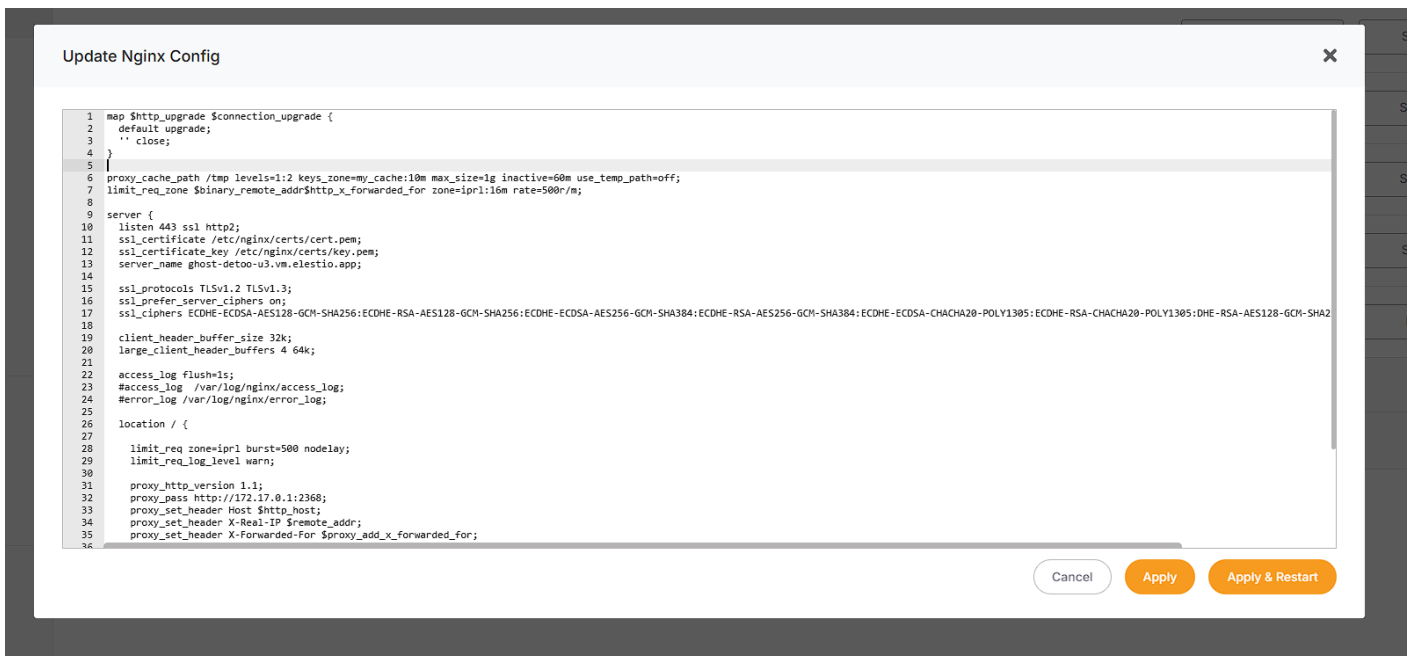
From the service overview screen it's possible to edit the nginx configuration for the elestio default subdomain domain & custom domain, to do that go to the Security tab, then click on Nginx configuration then click on Config button.



The screenshot displays the Elestio dashboard interface. On the left is a sidebar with the 'elestio' logo and a 'PROJECT: Elestio-services' dropdown. Below this is a list of navigation items: Services, CI/CD, Volumes, Load Balancer, Domains, Members, Billing, Project Setting, Audit Trail, Account, Support Tickets, Monitoring, and Documentation. The 'Services' item is highlighted. The main content area shows the 'ghost-detoo' service status as 'Running'. A horizontal tab bar includes Overview, Tools, Backups, Metrics, Monitoring, Logs, Audit, Security (selected), Alerts, and Notes. Under the 'Security' tab, there are sections for 'Enable/Disable Firewall', 'Rate Limiter', 'Output Cache', and 'Manage SSH Keys'. The 'Nginx Configuration' section contains a table with two columns: 'Cname' and 'Action'. One entry is listed with the cname 'ghost-detoo-u3.vm.elestio.app' and a 'Config' button in the action column.

Cname	Action
ghost-detoo-u3.vm.elestio.app	<a href="#">Config</a>

There you will see the site configuration in nginx, you are able from there to add or remove headers, to configure the reverse proxy and much more. Nginx is very powerful and provide tons of customizations and control.



## Edit the global Nginx configuration

It's also possible to edit the global nginx configuration, to do that follow those steps:

- 1) create a file named **nginx.conf** in **/opt/elestio/nginx/**
- 2) Paste exactly this content in the file, then you can customize it for your needs:

```
worker_processes auto;

worker_rlimit_nofile 100000;

events {
    worker_connections 4000;
}

http {
    include mime.types;
    default_type application/octet-stream;
    sendfile on;
    tcp_nopush on;
    tcp_nodelay on;
    server_tokens off;
```

```
# reasonable defaults
client_header_timeout 10m;
client_body_timeout 10m;
send_timeout 10m;
client_max_body_size 5120M;


connection_pool_size 256;
client_header_buffer_size 32k;
large_client_header_buffers 4 64k;
request_pool_size 4k;


server_names_hash_bucket_size 4096;


output_buffers 1 32k;
postpone_output 1460;


proxy_buffer_size 128k;
proxy_buffers 4 256k;
proxy_busy_buffers_size 256k;


ignore_invalid_headers on;


# enable gzip support
include server-gzip.conf;


# auto-ssl lua magic for automatic generation of certs
include resty-http.conf;


server {
    listen 80 default_server;


    include resty-server-http.conf;
}


include /etc/nginx/conf.d/*.conf;
}


stream {
    include /etc/nginx/streams/*.conf;
}
```

3) edit this file: **/opt/elestio/nginx/docker-compose.yml** and add there a new line under volumes

**- /opt/elestio/nginx/nginx.conf:/usr/local/openresty/nginx/conf/nginx.conf**

It should look like this:

```
version: '2'
services:
  nginx:
    image: elestio/nginx-auto-ssl:latest
    container_name: elestio-nginx
    restart: always
    network_mode: "host"
    volumes:
      - /opt/elestio/nginx/nginx.conf:/usr/local/openresty/nginx/conf/nginx.conf
      - /opt/elestio/nginx/ssl_data:/etc/ssl-auto-ssl
      - /opt/elestio/nginx/conf.d:/etc/nginx/conf.d
      - /opt/elestio/nginx/streams:/etc/nginx/streams
      - /opt/elestio/nginx/logs:/var/log/nginx/
      - /root/.acme.sh/vm.elestio.app/fullchain.cer:/etc/nginx/certs/cert.pem
      - /root/.acme.sh/vm.elestio.app/vm.elestio.app.key:/etc/nginx/certs/key.pem
    env_file:
      - .env
```

4) Restart nginx with this command:

```
cd /opt/elestio/nginx/;
docker-compose down;
docker-compose up -d;
```

Now nginx reverse proxy will use your custom global configuration

---

Revision #7

Created 6 March 2025 18:10:00 by Joseph Benguira

Updated 7 March 2025 09:02:41 by Joseph Benguira